

EXHIBIT B

UNREDACTED PUBLIC VERSION



IOENGINE, LLC. V PAYPAL HOLDINGS, INC.,

**IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE**

Case No. 1:18-cv-452 WCB

EXPERT REPORT OF DR. AVI RUBIN REGARDING VALIDITY OF U.S. PATENT NOS.
9,059,969 AND 9,774,703

January 7, 2022

Respectfully submitted,

Dr. Aviel D. Rubin.

**HIGHLY CONFIDENTIAL – ATTORNEYS’ EYES ONLY – SOURCE CODE
SUBJECT TO PROTECTIVE ORDER**

number, to a mapped network drive if a folder to be synchronized was located on that drive.” [Bims 899]. But Dr. Bims does not explain how such a process involves verification of the DiskOnKey itself. Further, as I explain in more detail below, Dr. Bims cannot prove that the alleged synchronization over a network is even possible with MyKey software. Dr. Bims has not tested the MyKey software. And nowhere in the description of the MyKey software does it say that the folder to be synchronized can be on a network. In fact the MyKey documentation states that the synchronization drive must be “*local*” to the host: “Select the folders you would like to synchronize. *The first folder must be an existing local folder.* The second folder must be an existing DiskOnKey folder.” [INGEN-0100391 at 12]. Indeed, the screenshot of the interface provided by Dr. Bims actually says, “**Host** Folder” and “DiskOnKey Folder” indicating that the folder with which to synchronize file contents is on the “**host** computer” to which the DiskOnKey is attached. [Bims 911]. Furthermore, Dr. Bims provides not explanation of what this “host computer” directory is. There is no reason to believe that “host computer” means anything other than exactly what it says—the computer that the DiskOnKey is connected to. Indeed, Figure 15, reproduced in Dr. Bims’ Report on page 449, shows an operation involving files on the “C:” drive, which is on the local host computer.

440. Dr. Bims also has not shown that there exists any fourth program code on the DiskOnKey which is configured to be executed by the portable device processor in response to a communication received by the portable device resulting from user interaction with the interactive user interface to initiate a synchronization operation, including one that facilitates verification of the portable device. Indeed, simple file copy operations do not necessarily require program code to be executed on a thumb drive like the DiskOnKey. Even assuming that the DiskOnKey was capable of synching files in an encrypted storage area, on USB devices the decryption operations

can be done entirely in hardware. In such a case, file copy operations are likely not controlled by software at all and thus Dr. Bims does not point to any program code running on the portable device during any folder synchronization operation. To the extent that Dr. Bims contends that firmware performs these operations, this is not necessarily true—many USB mass storage devices do not contain firmware. I am unaware of any hardware level analysis that Dr. Bims has performed to show the existence of code that implements any decryption operations. It is improper and unsupported to merely assume that decryption happens in software.

ii. The DiskOnKey Does Not Invalidate Claim 90

441. In my opinion, the DiskOnKey does not disclose “executing, in response to a communication received by the portable device resulting from user interaction with the interactive user interface, third program code stored on the portable device memory to cause a communication to be transmitted to a communications network node...[which] comprises providing the terminal with data stored on the portable device memory to facilitate the terminal to transmit a communication to the communications network node...wherein the data stored on the portable device memory comprises portable device identifier information.” [’703 Patent Claim 78, 86, 90]. Dr. Bims refers to “a number of example communications” that “were communications that were facilitated by the DiskOnKey providing identifier information stored on its memory to the terminal.” [Bims 921]. But the element of Claim 90 of the ’969 Patent requires “executing, *in response to a communication received by the portable device resulting from user interaction with the interactive user interface*, third program code stored on the portable device memory to cause a communication to be transmitted to a communications network node.” To the extent any of Dr. Bims’ examples do not involve the execution of code in response to a communication received by the portable device resulting from user interaction with the interactive user interface,

iii. The DiskOnKey Does Not Invalidate Claim 101

445. Dr. Bims treats Claim 101 of the '703 Patent as substantially identical to Claim 90, and does not rely on any arguments not already addressed in my discussion of Claim 90 above. [Bims 173-175]. As such Dr. Bims fails to disclose any opinions on how the DiskOnKey could anticipate Claim 101 of the '703 Patent. Regardless, as discussed in this report, the DiskOnKey does not anticipate any of the asserted claims.

iv. The DiskOnKey Does Not Invalidate Claim 105

446. Dr. Bims treats Claim 105 of the '703 Patent as substantially identical to Claim 56, and does not rely on any arguments not already addressed in my discussion of Claim 56 above. [Bims 173-175]. As such Dr. Bims fails to disclose any opinions on how the DiskOnKey could anticipate Claim 105 of the '703 Patent. Regardless, as discussed in this report, the DiskOnKey does not anticipate any of the asserted claims.

v. The DiskOnKey Does Not Invalidate Claim 114

447. First, I note that the DiskOnKey does not disclose the elements of Claim 104 of the '703 Patent, for substantially the same reasons as it does not disclose Claim 55 of the '703 Patent. Regardless, even if the DiskOnKey did disclose the elements of Claim 104 of the '703 Patent, which it does not, it does not disclose the additional element of Claim 114.

448. Dr. Bims first contends that during the DiskOnKey's alleged firmware update process, in response to user interaction on the computer, the DiskOnKey's processor is configured to execute code to cause a communication to be transmitted to the remote server "facilitate[s] synchronizing content on the portable device with content on the communications network node, such as the DiskOnKey's firmware version. [Bims 906-07]. I disagree.

449. First, as previously discussed, the DiskOnKey does not disclose any

communications in the alleged firmware update process that are the result of fourth program code which is configured to be executed by the portable device processor in response to a communication received by the portable device resulting from user interaction with the interactive user interface and configured to cause a communication to be transmitted to the communication network node.

450. Regardless, Dr. Bims fails to demonstrate that any alleged communications from the DiskOnKey to a server that are a part of a firmware update process are used to “facilitate[] synchronizing content on the [DiskOnKey] with content on the communication network node.”

451. First, Dr. Bims offers no proof that the “Checking for newer version” and “Getting new version” steps in the DiskOnKey Upgrade application refers to getting firmware or firmware information from a server. It may well be the case that the current firmware is locally stored on the terminal running the DiskOnKey Upgrade application and the “Check for newer version” and “Getting new version” steps refer to the *DiskOnKey* getting that new firmware from the terminal. Indeed, this would be equally possible because the very next alleged step is “Upgrading to new version,” and the DiskOnKey cannot upgrade to a version it does not have a copy of.

452. As a separate theory, Dr. Bims asserts that an alleged “synchronization” occurs through use of the MyKey application that discloses the element of Claim 10. I disagree for at least the following reasons.

453. First, Dr. Bims asserts only that “user interaction with the MyKey GUI” sends a communication to the DiskOnKey, in response to which the DiskOnKey causes information to be synchronized. [Bims 882]. But user interaction with a user interface to cause a network communication is not sufficient. Any alleged communication must be the result of fourth program code which is configured to be executed by the portable device processor in response to a

communication received by the portable device resulting from user interaction with the interactive user interface and configured to cause a communication to be transmitted to the communications network node. Dr. Bims simply fails to address these necessary elements of Claim 114. Moreover, Dr. Bims has not analyzed the alleged MyKey software and therefore does not know how it works or how it implements any alleged synchronization operations that are used. For instance, it is possible that the alleged synchronization component of MyKey software has already created a manifest for the contents of the DiskOnKey before any synchronization operations occur. In such a case it would just have to check the host folder to see what contents it has. Based on this check it could then determine what files to sync and thus communicate to the DiskOnKey to read/write as necessary. In this case, there is no interaction with the IUI that causes fourth program code to execute on the portable device to cause any communication to be sent.

454. Second, Dr. Bims cannot prove that the alleged synchronization over a network is even possible with MyKey software. Dr. Bims has not tested the MyKey software. And nowhere in the description of the MyKey software does it say that the folder to be synchronized can be on a network. In fact the MyKey documentation states that the synchronization drive must be “*local*” to the host: “Select the folders you would like to synchronize. *The first folder must be an existing local folder.* The second folder must be an existing DiskOnKey folder.” [INGEN-0100391 at 12]. Indeed, the screenshot of the interface provided by Dr. Bims actually says, “**Host** Folder” and “DiskOnKey Folder” indicating that the folder with which to synchronize file contents is on the “**host** computer” to which the DiskOnKey is attached. [Bims 911]. Furthermore, Dr. Bims provides not explanation of what this “host computer” directory is. There is no reason to believe that “host computer” means anything other than exactly what it says—the computer that the DiskOnKey is connected to. Indeed, Figure 15, reproduced in Dr. Bims’ Report on page 449,